Logistic Regression for Handwritten Digits Recognition

Sourabh Dilip Powar

February 26, 2018

Abstract

Logistic Regression uses the logistic function to find a model that fits with the data points. The function gives an 'S' shaped curve to model the data. The curve is restricted between 0 and 1, so it is easy to apply when y is binary. Logistic Regression can then model events better than linear regression, as it shows the probability for y being 1 for a given x value. Logistic Regression is used in statistics and machine learning to predict values of an input from previous test data.

1 Introduction

Logistic regression algorithm for Handwritten Digits Recognition takes training data consisting label and 16x16 pixel image. MNIST (Modified National Institute of Standards and Technology) handwritten digit database is used for both training and testing the learning algorithm. Input are the images of digit 1 and 5 with label 1 and 5 correspondingly. Algorithm plots calculate two features symmetry and intensity based on the input pixels. Learning algorithm learns weight and classify data based on features by using sigmoid function. The threshold is 0.5 used for the classification. .

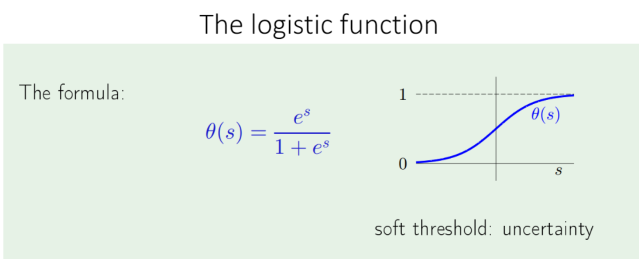

Figure 1: The sigmoid function for logistic regression.

Logistic regression algorithm

Initialize the weights at $t = 0$ to $\mathbf{w}(0)$
for $t = 0, 1, 2, \ldots$ do
  Compute the gradient

$$\nabla E_{\text{in}} = -\frac{1}{N}\sum_{n=1}^{N}\frac{y_n\mathbf{x}_n}{1+e^{y_n\mathbf{w}^{\intercal}(t)\mathbf{x}_n}}$$

  Update the weights: $\mathbf{w}(t+1) = \mathbf{w}(t) - \eta\nabla E_{\text{in}}$
  Iterate to the next step until it is time to stop
Return the final weights $\mathbf{w}$

Figure 2: The algorithm for logistic regression.

```
=== RESTART: E:\AI\ArtificialIntelligence\Assignment\HW2\HW2\code\main.py ===
logistic regression testing...
max iteration testcase0: Train accuracy: 0.975657, Test accuracy: 0.952830
max iteration testcase1: Train accuracy: 0.979500, Test accuracy: 0.962264
max iteration testcase2: Train accuracy: 0.979500, Test accuracy: 0.964623
max iteration testcase3: Train accuracy: 0.980141, Test accuracy: 0.959906
learning rate testcase0: Train accuracy: 0.979500, Test accuracy: 0.964623
learning rate testcase1: Train accuracy: 0.980141, Test accuracy: 0.959906
learning rate testcase2: Train accuracy: 0.981422, Test accuracy: 0.962264
logistic regression test done.
```

Figure 3: The Accuracy for logistic regression.

2 Implementation

2.1 Logistic regression

This function take the input data and learning rate. The weights are updated based on below algorithm.

2.2 Accuracy

Accuracy of both training and test data is calculated based on misclassified images. First 5 iteration will calculate accuracy by changing the number of iteration of learning algorithm and nest 5 cases will calculate based of varying learning rate. The experimental results for given training and test data are shown in Accuracy figure below.

2.3 Third order Transformation

This function converts the first order equation to third order equation by combination of the features such as symmetry and intensity. Their are together 10 features including biased input.

```
3rd order logistic regression testing...
max iteration testcase0: Train accuracy: 0.976297, Test accuracy: 0.957547
max iteration testcase1: Train accuracy: 0.978860, Test accuracy: 0.962264
max iteration testcase2: Train accuracy: 0.982063, Test accuracy: 0.959906
max iteration testcase3: Train accuracy: 0.982063, Test accuracy: 0.962264
learning rate testcase0: Train accuracy: 0.982063, Test accuracy: 0.957547
learning rate testcase1: Train accuracy: 0.982063, Test accuracy: 0.962264
learning rate testcase2: Train accuracy: 0.982703, Test accuracy: 0.959906
3rd order logistic regression test done.
```

Figure 4: The Accuracy for Third Order logistic regression.

2.4 Preference between linear model and third order transformation.

Third order transformation will go for over fitting if the weights assigned for higher order terms are considerable then the weights for low order. This will give high variance with the test data and will not be the desired output for the test cases. So we can use linear model or we should regularize the third order polynomial transformation while learning.

3 Conclusion and Results

The accuracy and result plot shows that we cannot achieve 100 percent. This is because the data set needed for the PLA classification must be linearly separable. Accuracy for the training and test dataset provided by proposed implementation are high for finite iterations taken on non linear separable dataset.